Attorney Docket No. 16869K-092500US
Client Ref. No. 593/SM/yk

PATENT APPLICATION

METHOD FOR UPDATING SECURITY INFORMATION, CLIENT, SERVER AND MANAGEMENT COMPUTER THEREFOR

Inventors: **Sastoshi OSHIMA**
Citizenship: Japan

Masahida SATO
Citizenship: Japan

Assignee: Hitachi, Ltd.
6, Kanda Surugadai 4-chome
Chiyoda-ku, Tokyo, Japan
Incorporation: Japan

Entity: Large

TOWNSEND AND TOWNSEND and CREW LLP
Two Embarcadero Center, 8th Floor
San Francisco, California 94111-3834
(415) 576-0200

METHOD FOR UPDATING SECURITY INFORMATION, CLIENT, SERVER AND MANAGEMENT COMPUTER THEREFOR

CROSS-REFERENCE TO RELATED APPLICATIONS

The present application claims priority upon Japanese Patent Application No. 2002-259190 filed on September 4, 2002, which is herein incorporated by reference.

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention relates to a method for updating security information, and to a client, a server and a management computer for use in the method.

Description of the Related Art

As shown in Fig. 7, a client that is connected to a server through a network comprises a CPU, a memory, a network interface, and a storage (local disk device) constituted by a hard disk device. An OS (Operating System), security software, and an application program are deployed in the memory. Various files are stored in the storage. These various files are a file group making up the OS, a policy file group, and a group of other files. The group of other files includes security software, an application program, data files (referred to simply as files as appropriate), and a

file system. Various file management information, such as respective file attributes and a file allocation table, are described in the file system (e.g., see Yasuharu Murase, Nyumon MS-DOS kaitei shinpan ("Introduction to MS-DOS, Newly Revised Edition"), ASCII Corporation, August 11, 1991, pp. 63-64).

As shown in Fig. 8, under the typical OS control, when the application program requests access to a file (S10), the file system is referenced by a file access control function, and it is determined whether or not access is to be permitted (S20). When access is permitted by this determination, the application program can access the file by a drive (S30 -> S40). The file that the application program accesses also includes files of the server connected through the network. That is, files of the server connected through the network are accessed by a network drive and a network card (S50 -> S60 -> S70).

Examples of the content of the file system stored in the storage include file attributes (see Fig. 2; e.g., see Yasuharu Murase, Nyumon MS-DOS kaitei shinpan ("Introduction to MS-DOS, Newly Revised Edition"), ASCII Corporation, August 11, 1991, pp. 131-132), an allocation table (see Fig. 3), and a cluster (see Fig. 4). As shown in the file attributes of Fig. 2, attributes such as file names, path names, owners, and group names are made to correspond to each file. Particularly in relation to access to the respective files by users, such as the owners, groups, and the outside, respective headings of read ("read" in Fig. 2), write ("write" in Fig. 2), and implementation are disposed.

Attributes relating to access columns of permitted ("permitted" in Fig. 2) and not permitted ("not permitted" in Fig. 2) are made to correspond to each of these headings.

The security software shown in Fig. 7 is software that is added to the OS and strengthens the access control function. The security software provides precise access control to further raise security on the basis of access control information described in the policy file. Detailed attributes, such as users, applications, and periods of time, are specified in the policy file as conditions permitting access to files.

A virus pattern file and a signature file are included in the policy file. The virus pattern file is used in virus countermeasure software called anti-virus software. The signature file is used in network attack countermeasure software called a host intrusion detection system. The anti-virus software inspects files, periodically or while a file is open, for the presence of computer virus infections on the basis of a pattern indicative of the characteristics of a computer virus, and appropriately takes necessary action. The host intrusion detection system identifies a network packet called a signature, detects attacks through the network, monitors e.g., a log file that the implemented application program outputs, and detects attacks.

The policy (security policy) referred to herein is security information generically naming limitations relating to use of a computer. In other words, the policy includes not only

information for prohibiting the implementation of specific applications and preventing changes to settings, but also various kinds of software.

Next, the updating of the policy file in the network environment will be described. As shown in Fig. 9, first, when power is turned on and the client starts up (S100), the OS thereof is activated (S110). Services resulting from a network connection and various pieces of software are started up (S120) by a control of the activated OS, whereby the transmission and reception of data via the network become possible. The updating function of the policy file is started up by the implementation of the security software (S130), and the client receives updates of the policy file from a management computer through the network. That is, the client receives the latest version of the policy file from the management computer (S140). Thus, the client is allowed to implement various processing corresponding to operational inputs by a user (S150).

In the above conventional technology, the policy file is stored in a storage equipped with the client. Thus, the policy file naturally cannot be updated while the operation of the client is stopped. For this reason, access control, virus countermeasures, etc., cannot be conducted in the latest state during the period of time until the policy file is updated.

As shown in the flow chart of Fig. 9, because the policy file of the client is updated after start-up, the client is operated with the old version of the security information during the period

of time from start-up until the security information is updated. In particular, network-infecting viruses attack devices connected to the network. Thus, the client is ordinarily in a state in which it is vulnerable to attack during the period of time from start-up until the security information is updated.

Moreover, in relation to updating of the policy, updating of the policy must be implemented with regard to each of a plurality of clients, and the entire management burden of each client becomes enormous.

SUMMARY OF THE INVENTION

An object of the present invention is to provide a method for updating security information even when operation of the client is stopped.

In order to accomplish the above and other objects, according to a first aspect of the present invention, there is provided a method for updating information on security , in which the client is connected with a server through a network, with the server including a storage device that is managed by the client, the storage device storing security information of the client, the method comprising updating the security information of the client stored in the storage device that the client manages in the server.

According to a second aspect of the present invention there is provided a client connected to a server through a network, the server including a storage device, comprising means for

managing the storage in the server, the storage device storing security information, the security information being updated without operation of the client, and means for referencing the security information of the client.

According to a third aspect of the present invention there is provided a server connected to a client through a network comprising means for communicating with the client through the network, and a storage device that is managed by the client, the storage device storing security information to be updated.

According to a fourth aspect of the present invention there is provided a management computer connected through a network to a server, the server including a storage device that is managed by a client, the storage device storing security information of the client, comprising means for communicating with the server through the network, and means for updating the security information.

BRIEF DESCRIPTION OF THE DRAWINGS

For a more complete understanding of the present invention and the advantages thereof, reference is now made to the following description taken in conjunction with the accompanying drawings wherein:

Fig. 1 is a system block diagram showing an embodiment according to the present invention;

Fig. 2 is a chart showing file attributes in the embodiment

according to the present invention;

Fig. 3 is a chart showing an allocation table in the embodiment according to the present invention;

Fig. 4 is a chart showing the structure of a cluster in the embodiment according to the present invention;

Fig. 5 is a flow chart showing a policy update sequence in the embodiment according to the present invention;

Fig. 6 is a flow chart showing additional access control of a file in the embodiment according to the present invention;

Fig. 7 is a block diagram showing the structure of a client in the embodiment according to the present invention;

Fig. 8 is a flow chart showing control of access to a file by a conventional client; and

Fig. 9 is a flow chart showing updating of a policy file by a conventional client.

DESCRIPTION OF THE PREFERRED EMBODIMENT

At least the following matters will be made clear by the explanation in the present specification and the description of the accompanying drawings.

A client is constituted by a so-called diskless computer that does not include a local hard disk device. In the present embodiment, this client will be called a "diskless client."

As shown in Fig. 1, one or a plurality of diskless clients 100 are connected through a network to a server 200 referred to

as a "storage server." A management computer 300 is connected through the network to the server 200. It should be noted that the management computer 300 may be omitted by having the server 200 include the function(s) of the management computer 300.

Similar to convention, the diskless client 100 is equipped with a CPU 110, a memory 120, and a network interface 130. An OS 121, security software 122, an application program 123, and a network storage driver 124 are read out from the server 200 and deployed in the memory 120. The diskless client 100 gives to the server 200 the function of a storage device as disk images 230.

That is, the diskless client 100 does not include a local disk device, but is mounted with a hard disk device (storage, storage device) of the server 200. The network storage driver 124 of the diskless client 100 controls the disk of the mounted server 200 via the network.

The server 200 includes a network interface 210 and a CPU 220, and also includes a plurality of hard disk devices that each diskless client 100 manages. The server 200 may also operate on, for example, a RAID (Redundant Array of Inexpensive Disks) format. The disk images 230 that the diskless client 100 uses are stored in the hard disk device. There is a disk image 230 present for each diskless client 100. The disk images 230 are file groups that are the same as the ones stored in the storage (hard disk device) of the conventional client shown in Fig. 7.

That is, the disk images 230 that the diskless client 100

uses are an OS configuration file group 231 relating to the configuration of the OS, a policy file group 232, and a group 233 of other files. Security information that is the same as that of the aforementioned conventional technology is included in these file groups 231 to 233.

That is, the policy file 232 is constituted by the aforementioned access control information, the virus pattern file, the signature file, and the like. The group 233 of other files includes the application program 123, data files that software and/or programs use, and the network storage driver 124. Additionally, various file management information shown in Figs. 2 to 4, such as file attributes and a file allocation table, is included as a file system in the group 233.

The management computer 300 includes a CPU 310, a memory 330, a network interface 320, and a storage 340 that is constituted by a hard disk device. An OS 331, management software 332, and an application program 333 are deployed in the memory 330. Various file groups 341 to 343, such as an OS, management software, an application program, and data files, are stored in the storage 340. The management software 332 uses the management software file group 342 to update the content of the various file groups 341 to 343 in the disk images 230 of the server 200, i.e., the security information, to update them to the latest versions.

Fig. 5 shows a specific updating sequence of the policy. As shown in Fig. 1, contrary to convention, the policy that the diskless client 100 uses is included in the various file groups

231 to 233 that the server 200 retains. As needed, the server 200 starts up the updating function of the policy (S200), appropriately receives the latest versions of the various file groups 231 to 233 from the management computer 300, and concludes the updating of the policy (S210). Thus, the updating of the policy, such as the file attributes, user ID, and various virus countermeasures, is conducted in the entire network to which the server 200 is connected. Thus, the latest security is secured with respect to unauthorized access and virus infiltration and attacks.

After updating of the policy has been concluded, the diskless client 100 starts up (S220). Specifically, when the diskless client 100 is turned on, an IPL (Initial Program Loader) is started up. A basic OS is called up from the disk images 230 of the server 200 through the network by the action of the IPL, and the basic OS is deployed in the memory 120 of the diskless client 100. The diskless client 100 begins operation as a computer by the deployed basic OS beginning operation (S230).

Services resulting from the network connection and various pieces of software are started by the control of the started OS (S240), whereby it becomes possible for the diskless client 100 to transmit and receive data through the network. When the diskless client 100 that has begun operation receives input of authentication information such as the ID (user name) and password of the user, it checks these with authentication information registered within the files groups 231 and 232 within the disk

images 230 of the server 200. When the validity of the authentication information including the group ID (group name) has been authenticated as a result of the checking, use of the client by the user is made possible.

Additional access control for raising security by the security software 122 will be described in a case where the application program is operated by the diskless client 100. The security software 122 is software added to the OS in order to strengthen the access control function. The security software 122 conducts precise access control for further raising security on the basis of the access control information described in the policy file. Detailed attributes, such as users, applications, and periods of time of day, are specified in the policy file as conditions permitting access to files that the application program and the like use.

That is, as shown in the flow chart of Fig. 6, when the application program 123 requests access to a file within the disk images 230 of the server 200 under the control of the OS (S310), additional file access control by the security software 122 functions (S311). The policy file 232 within the disk images 230 of the server 200 is referenced by the file access control, and the policy is authenticated with regard to the file(s) whose access has been requested (S312). The policy is information specifying the application, user ID permitting access, and attributes relating to each file. When file access by the application is permitted as a result of the authentication of the policy, next,

the file access control by the OS functions. The file system is referenced by the file access control of the OS with regard to the file(s) whose access has been requested, and the file attributes shown in Fig. 4 are verified (S313). When access is permitted by the user ID matching or the like as a result of this verification, the network storage drive 124 accesses the file(s) within the disk images 230 of the server 200 through the network interface 130 (S314 -> S315 -> S316).

Although an embodiment of the invention has been specifically described on the basis of that embodiment, the invention is not limited thereto and can be variously altered in a range that does not deviate from the gist thereof.

The following effects are provided with the embodiment of the invention.

The disk that the diskless client manages and uses is mounted on the server on the network. Additionally, the disk images of the client including the policy (security information) and files are stored in the disk mounted in the server. The management computer updates the policy in the disk images.

Thus, regardless of whether or not operation of the client is stopped, the policy (security information) can be updated as needed. Therefore, updating of the policy is already concluded at the point in time when the stopped client is started up. Thus, the client is operated according to the continually updated policy.

Even in a case where a plurality of clients mount disks on a single server, updating can be accomplished simply by updating

the policy of the server. For example, the policy of the server is updated through the management computer. That is, there is no need to implement updating of the policy with regard to each client as has conventionally been the case, and the management burden of each client can be greatly reduced.

Even when operation of the client is stopped, the security information can be updated.

Although the preferred embodiment of the present invention has been described in detail, it should be understood that various changes, substitutions and alterations can be made therein without departing from spirit and scope of the inventions as defined by the appended claims.